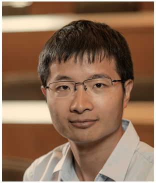
# Imitating Deep Learning Dynamics

## via Locally Elastic Stochastic Differential Equations



Jiayao Zhang[1,2] Hua Wang[1] Weijie J. Su[1]

[1] Department of Statistics [2] Cognitive Computation Group

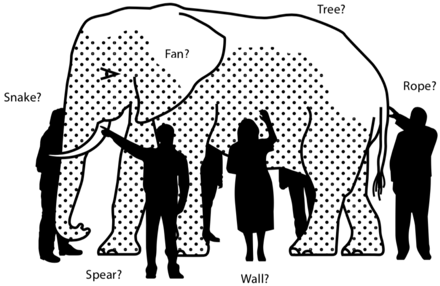
NeurIPS 2021

Penn
UNIVERSITY of PENNSYLVANIA

# Contents

# Motivation

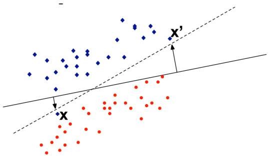
- A *phenomenological approach* for deep learning.
- We want
  - Big pictures instead of overly-complicated details;
  - Intuitive methods, though may not be fully rigorous without further work;
  - Guidance for future research toward demystifying deep models.

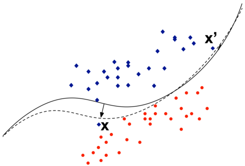



Penn
University *of* Pennsylvania

# Overview

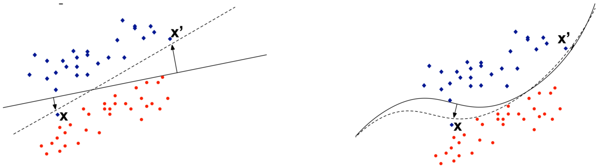
- Inspired by the *local elasticity* (LE, [HS20, DHS21, CHS20]) phenomenon: training on a sample *x* has a greater effect on samples that are similar to it than on those dissimilar to it.



# Overview

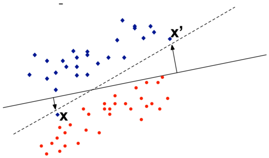
- Inspired by the *local elasticity* (LE, [HS20, DHS21, CHS20]) phenomenon: training on a sample *x* has a greater effect on samples that are similar to it than on those dissimilar to it.

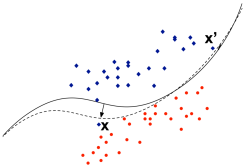



- How to encode this in our model?

# Overview

- Inspired by the *local elasticity* (LE, [HS20, DHS21, CHS20]) phenomenon: training on a sample $x$ has a greater effect on samples that are similar to it than on those dissimilar to it.



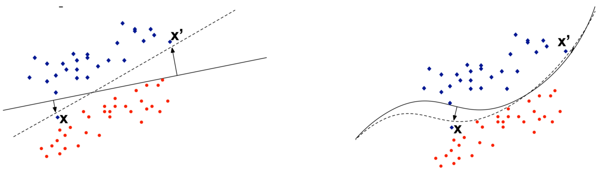

- How to encode this in our model?
- If at the $m$-th iteration, the $l$-th sample from the first class is trained, we model

$$\begin{cases} X_i^1(m) = X_i^1(m-1) + h \cdot \alpha X_l^1(m-1) + \text{noise}, \\ X_j^2(m) = X_j^2(m-1) + h \cdot \beta X_l^1(m-1) + \text{noise}. \end{cases} \quad (1)$$

Penn
University *of* Pennsylvania

# Overview

- Inspired by the *local elasticity* (LE, [HS20, DHS21, CHS20]) phenomenon: training on a sample $x$ has a greater effect on samples that are similar to it than on those dissimilar to it.




- How to encode this in our model?

- If at the $m$-th iteration, the $l$-th sample from the first class is trained, we model

$$\begin{cases} X_i^1(m) = X_i^1(m-1) + h \cdot \alpha X_l^1(m-1) + \text{noise}, \\ X_j^2(m) = X_j^2(m-1) + h \cdot \beta X_l^1(m-1) + \text{noise}. \end{cases} \quad (1)$$

- Then the emergence of LE can be understood as $\gamma := \alpha - \beta$ being large.

Penn
University *of* Pennsylvania

# Model Overview (1/2)

- **The LE-SDE: modeling feature dynamics with LE.**

$$\mathrm{d}\tilde{\boldsymbol{X}}(t) = \boldsymbol{M}(t)\tilde{\boldsymbol{X}}(t)\,\mathrm{d}t + \boldsymbol{\Sigma}(t)\,\mathrm{d}\boldsymbol{B}_t, \tag{2}$$

where $\tilde{\boldsymbol{X}}(t) = (\tilde{X}^k(t))_{k=1}^K \in \mathbb{R}^{Kp}$ is the concatenation of $p$-dimensional feature vectors from $K$ classes. We model the drift

$$\boldsymbol{M}(t) = (\boldsymbol{E}(t) \otimes \boldsymbol{P}) \circ \boldsymbol{H} \tag{3}$$

where the LE matrix $\boldsymbol{E}(t) \in \mathbb{R}^{K\times K}$ models the strength of LE, the sampling matrix $\boldsymbol{P} \in \mathbb{R}^{K\times K}$ models sampling effects, and a "similarity matrix" $\boldsymbol{H} \in \mathbb{R}^{Kp\times Kp}$ (as a $K$-by-$K$ block matrix) that models the direction features interacts under LE.

The simplest LE matrix can be set to be one with $\alpha(t)$ (intra-class effects) on its diagonal and $\beta(t)$ (inter-class effects) elsewhere.

Penn
University of Pennsylvania

# Model Overview (2/2)

- **The LE-ODE: dynamics on mean features $\bar{X}$.**

$$\mathrm{d}\bar{\boldsymbol{X}}(t) = \boldsymbol{M}(t)\bar{\boldsymbol{X}}(t)\,\mathrm{d}t = \left(\left(\boldsymbol{E}(t) \otimes \boldsymbol{P}\right) \circ \boldsymbol{H}\right)\bar{\boldsymbol{X}}(t)\,\mathrm{d}t. \tag{4}$$

E.g., given $\boldsymbol{P} = \mathbf{1}_{K\times K}/K$ and the two-parameter LE $\boldsymbol{E}(t)$,

$$\mathrm{d}\underbrace{\begin{bmatrix}\begin{bmatrix}\bar{X}_1^1\\ \vdots\\ \bar{X}_p^1\end{bmatrix}\\ \vdots\\ \begin{bmatrix}\bar{X}_1^K\\ \vdots\\ \bar{X}_p^K\end{bmatrix}\end{bmatrix}}_{\bar{\boldsymbol{X}}(t)} = \frac{1}{K}\underbrace{\begin{bmatrix}\alpha(t) & \beta(t) & \dots & \beta(t)\\ \beta(t) & \alpha(t) & \dots & \beta(t)\\ \vdots & \vdots & \dots & \vdots\\ \beta(t) & \dots & \dots & \alpha(t)\end{bmatrix}}_{\boldsymbol{E}(t)} \circ \underbrace{\begin{bmatrix}\boldsymbol{H}_{11} & \boldsymbol{H}_{12} & \dots & \boldsymbol{H}_{1K}\\ \boldsymbol{H}_{21} & \boldsymbol{H}_{22} & \dots & \boldsymbol{H}_{2K}\\ \vdots & \dots & \dots & \vdots\\ \boldsymbol{H}_{K1} & \boldsymbol{H}_{K2} & \dots & \boldsymbol{H}_{KK}\end{bmatrix}}_{\boldsymbol{H}} \begin{bmatrix}\begin{bmatrix}\bar{X}_1^1\\ \vdots\\ \bar{X}_p^1\end{bmatrix}\\ \vdots\\ \begin{bmatrix}\bar{X}_1^K\\ \vdots\\ \bar{X}_p^K\end{bmatrix}\end{bmatrix}\mathrm{d}t. \tag{5}$$

# Main Results

- **Separation Theorem:** features are asymptotically linearly separable if there is LE ("$\alpha(t) > \beta(t)$") for PSD $\boldsymbol{H}$ with positive diagonals.
- **Modeling choices for $\boldsymbol{H} = (\boldsymbol{H}_{ij})_{ij}$ matrix.**

| Model | $\boldsymbol{H}_{ij}$ | Remark |
|---|---|---|
| I-model | $\boldsymbol{I}_p$ | Isotropic Feature Model |
| L-model | $\bar{\boldsymbol{H}}^j = \boldsymbol{d}_j\boldsymbol{d}_j^\top / \|\boldsymbol{d}_j\|_2^2$ | Logits-as-Features Model |

Table 1: Modeling choices for $\boldsymbol{H}$, where $\boldsymbol{d}_j = \boldsymbol{e}_j - \frac{1}{K}\mathbf{1}_p$ for $j \in [K]$.

- **Simulating genuine dynamics with the LE matrix estimated.**

Penn
University *of* Pennsylvania

# Contents



University of Pennsylvania

# The Separation Theorem

## Theorem (Separation of LE-SDE)

*Suppose $\gamma(t) = \alpha(t) - \beta(t) > 0$, assume $\boldsymbol{H} = (\boldsymbol{H}_{ij})_{ij}$ is positive semi-definite (PSD) with positive diagonal entries. As $t \to \infty$, we have*

1. *if $\gamma(t) = \omega(1/t)$, the features are separable with probability tending to $1$;*
2. *if $\gamma(t) = o(1/t)$, and the number of per-class-feature $n$ tending to $\infty$ at an arbitrarily slow rate, the features are asymptotically pairwise separable with probability $0$.*

Here, $\gamma(t) = \omega(1/t)$ stands for $\gamma(t) \gg 1/t$ as $t \to \infty$. For example, $1/t^{0.5} = \omega(1/t)$ and $(t \ln t)^{-1} = o(1/t)$ as $t \to \infty$.

University of Pennsylvania

# Proof Sketch

- Substituting back the solution of the LE-ODE

$$\bar{\boldsymbol{X}}_t = \bar{\boldsymbol{X}}_0 + \sum_{i=1}^{Kp} c_i \boldsymbol{u}_i \, \mathrm{e}^{\mu_i t}, \quad \bar{\boldsymbol{X}}_0 = \sum_{i=1}^{Kp} c_i \boldsymbol{u}_i, \tag{6}$$

to the LE-SDE, we have

$$\begin{aligned} \tilde{\boldsymbol{X}}^k(t) &= \tilde{\boldsymbol{X}}^k(0) + \boldsymbol{M}_t \bar{\boldsymbol{X}}(t) - \mathbb{E}[\tilde{\boldsymbol{X}}^k(0)] + \boldsymbol{\Sigma}_k^{\frac{1}{2}}(t) \boldsymbol{W}^k(t) \\ &= \tilde{\boldsymbol{X}}^k(0) + \sum_{i=1}^{Kp} c_i \mu_i \boldsymbol{u}_i^k \, \mathrm{e}^{\mu_i t} - \sum_{i=1}^{Kp} c_i \boldsymbol{u}_i^k + \boldsymbol{\Sigma}_k^{\frac{1}{2}} \boldsymbol{W}^k(t), \end{aligned} \tag{7}$$

- To prove separation, it suffices to identify a direction $\boldsymbol{\nu}$ such that

$$\left\langle \tilde{\boldsymbol{X}}^k(t) - \tilde{\boldsymbol{X}}^l(t), \boldsymbol{\nu} \right\rangle > 0, \quad \text{w.p.} \rightarrow 1 \text{ as } t \rightarrow \infty, \quad \forall k \neq l. \tag{8}$$

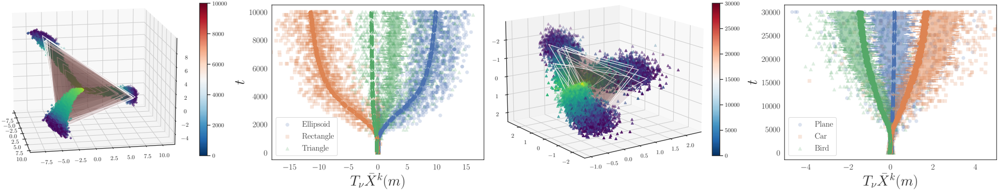
- Using Gaussian tail bound to obtain the rates; using nullity theorems to show $\boldsymbol{\nu}$ can be chosen independent of the class indices.

# Corollary

Neural collapse [PHD20, FHLS21] is a recent phenomenological finding on the geometry of logits of DNNs at convergence: they tend to form equiangular tight frames (ETFs).

## Proposition (Neural Collapse of the LE-ODE)

*Under L-model and the same setup as in Theorem 1, if $\gamma(t) > 0$ and there exists some $T > 0$ such that $B(t) < 0$ for $t \geq T$, then $\left\{\bar{\boldsymbol{X}}^k(t)/\|\bar{\boldsymbol{X}}^k(t)\|\right\}_{k=1}^{K}$ forms an ETF as $t \to \infty$.*




(a) GeoMNIST in $\mathbb{R}^3$. (b) GeoMNIST in $\mathbb{R}$. (c) CIFAR in $\mathbb{R}^3$. (d) CIFAR in $\mathbb{R}$.

# Justifications for Linearization (1/5)

- **The genuine dynamics of logits.**

$$\boldsymbol{X}_i^k(m) - \boldsymbol{X}_i^k(m-1) \approx h \left[ \frac{\partial \boldsymbol{X}_i^k(m-1)}{\partial \boldsymbol{w}} \frac{\partial \boldsymbol{X}_{J_m}^{L_m}}{\partial \boldsymbol{w}}^\top \left( \boldsymbol{e}_{L_m} - \mathrm{softmax}(\boldsymbol{X}_{J_m}^{L_m}) \right) \right]. \tag{9}$$

- **First approximation: decoupling in an expectation.**

$$\begin{aligned} \mathrm{d}\tilde{\boldsymbol{X}}_t^k &\approx \mathbb{E}_{L \sim \mathrm{U}([K])} \left[ \mathbb{E}_{\tilde{\boldsymbol{X}} \sim \mathcal{D}_t^L} \left[ \frac{\partial \boldsymbol{X}_i^k(m-1)}{\partial \boldsymbol{w}} \frac{\partial \tilde{\boldsymbol{X}}}{\partial \boldsymbol{w}}^\top \left( \boldsymbol{e}_L - \mathrm{softmax}(\tilde{\boldsymbol{X}}) \right) \right] \right] \mathrm{d}t + \boldsymbol{\Sigma}_t^{\frac{1}{2}} \, \mathrm{d}\boldsymbol{W}_t, \\ &\approx \frac{1}{K} \sum_L \left( \left[ \mathbb{E}_{\tilde{\boldsymbol{X}}' \sim \mathcal{D}_t^k, \tilde{\boldsymbol{X}} \sim \mathcal{D}_t^L} \frac{\partial \tilde{\boldsymbol{X}}'}{\partial \boldsymbol{w}} \frac{\partial \tilde{\boldsymbol{X}}}{\partial \boldsymbol{w}}^\top \right] \left( \boldsymbol{e}_L - \mathrm{softmax}(\bar{\boldsymbol{X}}_t^L) \right) \right) \mathrm{d}t + \boldsymbol{\Sigma}_t^{\frac{1}{2}} \, \mathrm{d}W_t, \\ &= \frac{1}{K} \sum_L \left( \Theta_{k,L} \left( \boldsymbol{e}_L - \mathrm{softmax}(\bar{\boldsymbol{X}}_t^L) \right) \right) \mathrm{d}t + \boldsymbol{\Sigma}_t^{\frac{1}{2}} \, \mathrm{d}\boldsymbol{W}_t. \end{aligned} \tag{10}$$

Penn
University of Pennsylvania

# Justifications for Linearization (2/5)

- **Linearize the drift $F$ around the mean at each time.**

$$F(\tilde{\boldsymbol{X}}(t), t) := \Theta(t) \left( \left[ e_k - \sigma(\tilde{\boldsymbol{X}}^k(t)) \right]_{k=1}^{K} \right), \tag{11}$$

$$F(\tilde{\boldsymbol{X}}(t), t) \approx \tilde{F}(\tilde{\boldsymbol{X}}(t), t) := F(\varphi(t), t) + \nabla_X F(\varphi(t), t) \left( \tilde{\boldsymbol{X}}(t) - \varphi(t) \right), \tag{12}$$

where $\varphi(t) := \bar{\boldsymbol{X}}(t)$, $J = \nabla_X F = J$ is a block diagonal matrix $J = (J_{kk})$ with $J_{kk} = J_k := \mathrm{diag}(\bar{p}_k) - \bar{p}_k \bar{p}_k^T$, here we write

$$p = (p_k)_{k=1}^{K} \in \mathbb{R}^{Kp}, \quad p_k := \sigma(\tilde{\boldsymbol{X}}^k(t)) \in \mathbb{R}^p, \quad k \in [K], \tag{13}$$

and similarly

$$\bar{p} = (\bar{p}_k)_{k=1}^{K} \in \mathbb{R}^{Kp}, \quad \bar{p}_k := \sigma(\bar{\boldsymbol{X}}^k(t)) \in \mathbb{R}^p, \quad k \in [K]. \tag{14}$$

Penn
University *of* Pennsylvania

# Justifications for Linearization (3/5)

- **Linearize the drift $F$ around the mean at each time (cont'd).**

$$\begin{aligned}\tilde{F}(\tilde{\boldsymbol{X}}(t), t) &= \Theta(t)\left([e_k - \bar{p}_k]_k + J(t)(\tilde{\boldsymbol{X}}(t) - \varphi(t))\right) \\ &= \Theta(t)\left(J(t)X(t) + [e_k - \bar{p}_k + J_k\varphi_k(t)]_k\right).\end{aligned} \tag{15}$$

Define $\Psi : \mathbb{R}^{Kp} \to \mathbb{R}^{Kp} : z \mapsto [e_k - \sigma(z_k)]_k$ and write $\Psi_k : \mathbb{R}^p \to \mathbb{R}^p$ to be the $k$-th component of $\Psi$, expand $\Psi(z)$ around $\varphi(t)$ for each $t$:

$$\Psi = \Psi(\varphi) + J(t)\varphi - J(\varphi)z + o\left(\|z - \varphi\|\right), \tag{16}$$

or

$$\Psi(\varphi) + J(t)\varphi = \Psi(z) + J(\varphi)z + o\left(\|z - \varphi\|\right). \tag{17}$$

This implies that

$$\tilde{F} = \Theta(t)J(t)\tilde{\boldsymbol{X}}(t) + \Theta(t)R(t), \quad R(t;z) := \Psi(z) + J(t)z + o\left(\|z - \varphi(t)\|\right). \tag{18}$$

Penn
University *of* Pennsylvania

# Justifications for Linearization (4/5)

- **Point $z$ for expansion.**
  - **Around initialization: constant residue.** Let $z = u := c \cdot [\mathbf{1}_K/K]_{k=1}^{K}$ be a scaling of vectors of ones where $c$ is some fixed constant. Then each of the $K$ components of $\sigma(u)$ assigns approximately the same probability $(1/K)$ for every label. Furthermore, $u \in \operatorname{Ker} J(t)$ for all $t$ hence the residue $R(t; u) = \Psi(u) + o\left(\|z - \varphi(t)\|\right)$ is a constant vector.
  - **Around convergence: vanishing residue.** Given that the model converges, $\varphi_\infty := \varphi(\infty)$ is finite. Let $z = \varphi_\infty$, under the effective training assumption, $\|\Psi(\varphi_\infty)\| \approx 0$ by construction. Hence the residue $R(t; \varphi_\infty) = J(t)\varphi_\infty + o\left(\|\varphi(t) - \varphi_\infty)\|\right)$. Here the $o(\cdot)$ term converges to $0$ as training progresses, leaving us a term that is asymptotically equivalent to $v = (v_k)_{k=1}^{K} := J(\varphi_\infty)\varphi_\infty \in \mathbb{R}^{K^2}$, where $v_k = [(z_{k,i} - \sum_{j=1}^{K} p_{k,j} z_{k,j})p_i]_{i=1}^{K} \in \mathbb{R}^K \approx \mathbf{0}_K$ under the effective training assumption. In this regime, the residue $o\left(\|\varphi(t) - \varphi_\infty\|\right)$ eventually vanishes.

Penn
University *of* Pennsylvania

# Justifications for Linearization (5/5)

**Summary of Approximations**

- Decoupling inside an expectation.
- Linearize the drift around the mean $\bar{\boldsymbol{X}}$.
- First-order expansion around convergence.

# Contents

# Setup

- **Datasets and Models.**

# Setup

- **Datasets and Models.**
  - GeoMNIST: $K = 3$ classes of simple geometric shapes (Rectangle, Ellipsoid, and Triangle).

# Setup

- **Datasets and Models.**
  - GeoMNIST: $K = 3$ classes of simple geometric shapes (Rectangle, Ellipsoid, and Triangle).
  - CIFAR-10: $5000$ training samples and $1000$ validation samples per class, with the total number of classes $K \in [2, 3]$.

# Setup

- **Datasets and Models.**
  - GeoMNIST: $K = 3$ classes of simple geometric shapes (Rectangle, Ellipsoid, and Triangle).
  - CIFAR-10: $5000$ training samples and $1000$ validation samples per class, with the total number of classes $K \in [2, 3]$.
- **Training Configurations.**

# Setup

- **Datasets and Models.**
  - GeoMNIST: $K = 3$ classes of simple geometric shapes (Rectangle, Ellipsoid, and Triangle).
  - CIFAR-10: $5000$ training samples and $1000$ validation samples per class, with the total number of classes $K \in [2, 3]$.
- **Training Configurations.**
  - Variants of the AlexNet model ([KSH12]): two convolutional layers and three fully-connected layers activated by ReLU.

# Setup

- **Datasets and Models.**
  - GeoMNIST: $K = 3$ classes of simple geometric shapes (Rectangle, Ellipsoid, and Triangle).
  - CIFAR-10: $5000$ training samples and $1000$ validation samples per class, with the total number of classes $K \in [2, 3]$.
- **Training Configurations.**
  - Variants of the AlexNet model ([KSH12]): two convolutional layers and three fully-connected layers activated by $\texttt{ReLU}$.
  - All models are trained for $T = 10^5$ iterations (for GeoMNIST) or $T = 3 \times 10^5$ iterations (for CIFAR) with a learning rate of $0.005$ and a batch size of $1$ under the softmax cross-entropy loss. Models on GeoMNIST converged with training and validation losses to zero, and those on CIFAR to validation accuracies greater than $90\%$.

Penn
UNIVERSITY *of* PENNSYLVANIA

# LE Coefficients Estimation

- **Estimation Procedure.** Define $A(t) = \int_0^t \alpha(\tau)\,\mathrm{d}\tau$, $B(t) = \int_0^t \beta(\tau)\,\mathrm{d}\tau$, write out exact solutions under the I-model and the L-model, we can estimate

$$
\begin{aligned}
&\text{(I-model)} \quad \begin{cases} \hat{A}(t) &= \operatorname{avg} \operatorname{avg}_k \log \left| \frac{\check{\boldsymbol{x}}(\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}})^{K-1}}{\boldsymbol{c}_0 \boldsymbol{c}_k^{K-1}} \right|, \\ \hat{B}(t) &= -\operatorname{avg} \operatorname{avg}_k \log \left| \frac{\boldsymbol{c}_0}{\boldsymbol{c}_k} \frac{\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}}}{\check{\boldsymbol{x}}} \right|, \end{cases} \quad \check{\boldsymbol{X}}_t := \operatorname{avg}_l \bar{\boldsymbol{X}}_t^l, \\
&\text{(L-model)} \quad \begin{cases} \hat{A}(t) &= A'(t) + 2B'(t), \\ \hat{B}(t) &= 2(B'(t) - A'(t)), \end{cases} \quad \begin{cases} A'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \boldsymbol{v}_1 - 1 \right\rangle \right|, \\ B'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \left( \boldsymbol{v}_2 - \frac{4}{3} \boldsymbol{v}_1 \right) \right\rangle \right|, \end{cases}
\end{aligned} \tag{19}
$$

where $\operatorname{avg}_l(\cdot)$ denotes averaging over axis $l$, and $\operatorname{avg}(\cdot)$ averaging all elements.

# LE Coefficients Estimation

- **Estimation Procedure.** Define $A(t) = \int_0^t \alpha(\tau)\,\mathrm{d}\tau$, $B(t) = \int_0^t \beta(\tau)\,\mathrm{d}\tau$, write out exact solutions under the I-model and the L-model, we can estimate

$$
\begin{aligned}
&(\text{I-model}) \quad \begin{cases} \hat{A}(t) &= \operatorname{avg}\operatorname{avg}_k \log\left|\frac{\check{\boldsymbol{x}}(\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}})^{K-1}}{c_0 c_k^{K-1}}\right|, \\ \hat{B}(t) &= -\operatorname{avg}\operatorname{avg}_k \log\left|\frac{c_0}{c_k}\frac{\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}}}{\check{\boldsymbol{x}}}\right|, \end{cases} \quad \check{\boldsymbol{X}}_t := \operatorname{avg}_l \bar{\boldsymbol{X}}_t^l, \\
&(\text{L-model}) \quad \begin{cases} \hat{A}(t) &= A'(t) + 2B'(t), \\ \hat{B}(t) &= 2(B'(t) - A'(t)), \end{cases} \quad \begin{cases} A'(t) &:= \log\left|\left\langle \bar{\boldsymbol{X}}^\top \boldsymbol{v}_1 - 1\right\rangle\right|, \\ B'(t) &:= \log\left|\left\langle \bar{\boldsymbol{X}}^\top \left(\boldsymbol{v}_2 - \frac{4}{3}\boldsymbol{v}_1\right)\right\rangle\right|, \end{cases}
\end{aligned}
\tag{19}
$$

where $\operatorname{avg}_l(\cdot)$ denotes averaging over axis $l$, and $\operatorname{avg}(\cdot)$ averaging all elements.

- Main idea: eigenvectors of the $Kp$-by-$Kp$ drift matrix $M(t)$ as concatenations of $K$ vectors in $\mathbb{R}^p$ and construct their linear combinations such that one or more independent components in the solution vanishes.

Penn
University *of* Pennsylvania

# LE Coefficients Estimation

- **Estimation Procedure.** Define $A(t) = \int_0^t \alpha(\tau)\,\mathrm{d}\tau$, $B(t) = \int_0^t \beta(\tau)\,\mathrm{d}\tau$, write out exact solutions under the I-model and the L-model, we can estimate

$$
\begin{aligned}
&(\text{I-model}) \quad \begin{cases} \hat{A}(t) &= \operatorname{avg} \operatorname{avg}_k \log \left| \dfrac{\check{\boldsymbol{x}}(\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}})^{K-1}}{\boldsymbol{c}_0 \boldsymbol{c}_k^{K-1}} \right|, \\ \hat{B}(t) &= -\operatorname{avg} \operatorname{avg}_k \log \left| \dfrac{\boldsymbol{c}_0}{\boldsymbol{c}_k} \dfrac{\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}}}{\check{\boldsymbol{x}}} \right|, \end{cases} \quad \check{\boldsymbol{X}}_t := \operatorname{avg}_l \bar{\boldsymbol{X}}_t^l, \\
&(\text{L-model}) \quad \begin{cases} \hat{A}(t) &= A'(t) + 2B'(t), \\ \hat{B}(t) &= 2(B'(t) - A'(t)), \end{cases} \quad \begin{cases} A'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \boldsymbol{v}_1 - 1 \right\rangle \right|, \\ B'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \left( \boldsymbol{v}_2 - \frac{4}{3} \boldsymbol{v}_1 \right) \right\rangle \right|, \end{cases}
\end{aligned}
\tag{19}
$$

  where $\operatorname{avg}_l(\cdot)$ denotes averaging over axis $l$, and $\operatorname{avg}(\cdot)$ averaging all elements.

- Main idea: eigenvectors of the $Kp$-by-$Kp$ drift matrix $M(t)$ as concatenations of $K$ vectors in $\mathbb{R}^p$ and construct their linear combinations such that one or more independent components in the solution vanishes.

- Obtain $\hat{\alpha}(t)$ and $\hat{\beta}(t)$ using the Savitzky-Golay filter.

Penn
University *of* Pennsylvania

# LE Coefficients Estimation

- **Estimation Procedure.** Define $A(t) = \int_0^t \alpha(\tau)\,\mathrm{d}\tau$, $B(t) = \int_0^t \beta(\tau)\,\mathrm{d}\tau$, write out exact solutions under the I-model and the L-model, we can estimate

$$
\begin{aligned}
&\text{(I-model)} \quad \begin{cases} \hat{A}(t) &= \operatorname{avg}\operatorname{avg}_k \log \left| \frac{\check{\boldsymbol{x}}(\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}})^{K-1}}{c_0 c_k^{K-1}} \right|, \\ \hat{B}(t) &= -\operatorname{avg}\operatorname{avg}_k \log \left| \frac{c_0}{c_k} \frac{\bar{\boldsymbol{x}}^k - \check{\boldsymbol{x}}}{\check{\boldsymbol{x}}} \right|, \end{cases} \quad \check{\boldsymbol{X}}_t := \operatorname{avg}_l \bar{\boldsymbol{X}}_t^l, \\
&\text{(L-model)} \quad \begin{cases} \hat{A}(t) &= A'(t) + 2B'(t), \\ \hat{B}(t) &= 2(B'(t) - A'(t)), \end{cases} \quad \begin{cases} A'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \boldsymbol{v}_1 - 1 \right\rangle \right|, \\ B'(t) &:= \log \left| \left\langle \bar{\boldsymbol{X}}^\top \left( \boldsymbol{v}_2 - \frac{4}{3}\boldsymbol{v}_1 \right) \right\rangle \right|, \end{cases}
\end{aligned} \tag{19}
$$

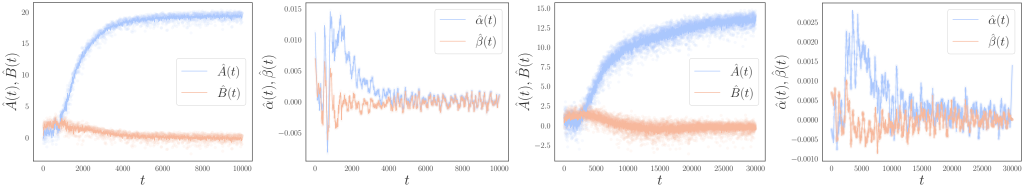
  where $\operatorname{avg}_l(\cdot)$ denotes averaging over axis $l$, and $\operatorname{avg}(\cdot)$ averaging all elements.

- Main idea: eigenvectors of the $Kp$-by-$Kp$ drift matrix $M(t)$ as concatenations of $K$ vectors in $\mathbb{R}^p$ and construct their linear combinations such that one or more independent components in the solution vanishes.

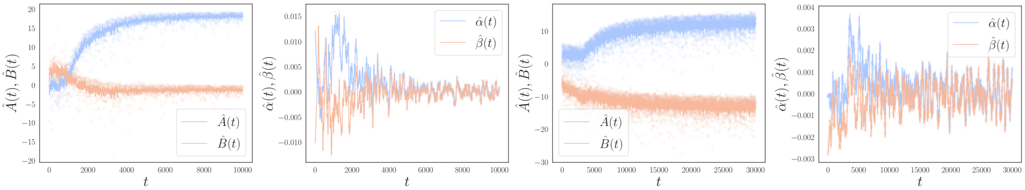
- Obtain $\hat{\alpha}(t)$ and $\hat{\beta}(t)$ using the Savitzky-Golay filter.

- *Tail index* $r_\alpha := \sup_s \{s : \lim_{t\to\infty} \alpha(t) \cdot t^s < \infty\}$, estimated by $\hat{r}_\alpha = 1 - \operatorname{avg}_{T-1000 \le t \le T} \frac{\log \alpha(t)}{\log(1+t)}$.

# LE Coefficient Estimation




(a) GeoMNIST (I-model). (b) GeoMNIST (I-model). (c) CIFAR (I-model). (d) CIFAR (I-model).

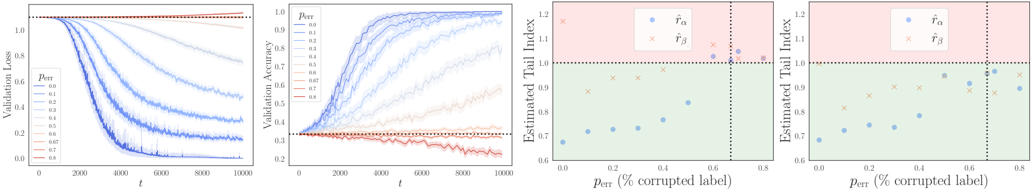



(e) GeoMNIST (L-model). (f) GeoMNIST (L-model). (g) CIFAR (L-model). (h) CIFAR (L-model).

Figure 2: **Estimated $\hat{A}(t)$, $\hat{B}(t)$, $\alpha(t)$, and $\beta(t)$.** The first row was estimated using I-model and the second L-model; the first two columns are on GeoMNIST and the last two on CIFAR. The first and third rows show $\hat{A}(t)$ and $\hat{B}(t)$ and the other two rows $\hat{\alpha}(t)$ and $\hat{\beta}(t)$.

Penn
UNIVERSITY *of* PENNSYLVANIA

# Verifying the Separation Theorem

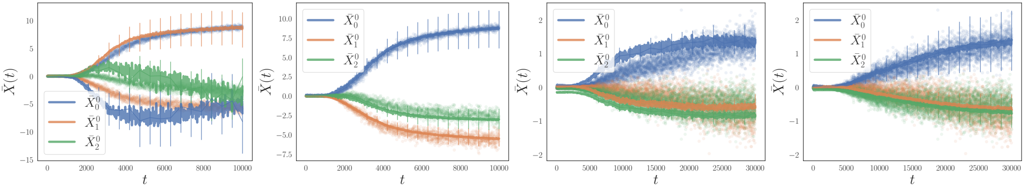



(a) Validation loss. (b) Validation accuracy. (c) I-model. (d) L-model.

Figure 3: **Phase transition of separability over label pollution ratio $p_{\text{err}}$. (a)—(b)** Validation loss and accuracy suggest separation fails for $p_{\text{err}} \geq p^*_{\text{err}} = 2/3$. The dashed line in (a) carries the value at initialization and overlaps with the case where $p_{\text{err}} = 0.6$; the dashed line in (b) is $p^*_{\text{err}} = 2/3$, when labels are assigned completely at random. **(c)—(d)** Tail indices of $\alpha(t)$ and $\beta(t)$ estimated using the I-model and L-model resp. Although the case for the L-model does not exhibit a clear phase transition, we note around $p_{\text{err}} \approx 2/3$, the tail index of $\hat{\beta}(t)$ begins to dominate that of $\hat{\alpha}(t)$.

Penn
UNIVERSITY of PENNSYLVANIA

# Simulating Dynamics via LE-SDE



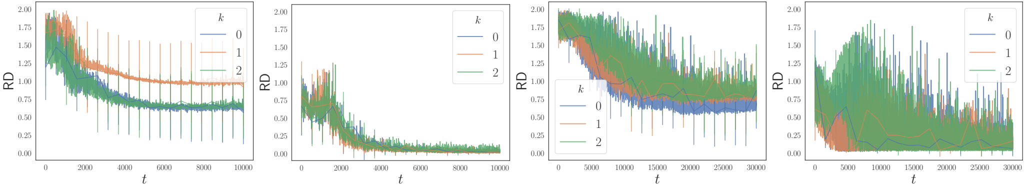

(a) GeoMNIST (I-model).　(b) GeoMNIST (L-model).　(c) CIFAR (I-model).　(d) CIFAR (L-model).

Figure 4: **Simulated LE-ODE solutions versus genuine dynamics.** We use $\hat{\alpha}(t)$ and $\hat{\beta}(t)$ estimated from I-model ((a) and (c)) or L-model, ((b) and (d)) and numerically simulate the solution under the L-model. The results were overlaid with true dynamics from neural nets. We note L-model in general imitated true dynamics reasonably well.

# Residues of Simulating Dynamics via LE-SDE

We measure the goodness-of-fit via relative difference (RD, the lower the better) defined for each class $k \in [K]$ as

$$\mathrm{RD_k}(t) := \frac{\left\|\bar{\boldsymbol{X}}^k(t) - \bar{\boldsymbol{Y}}^k(t)\right\|_{\boldsymbol{H}^k}}{\left(\left\|\bar{\boldsymbol{X}}^k(t)\right\|_2 + \left\|\bar{\boldsymbol{Y}}^k(t)\right\|_2\right)/2}, \tag{20}$$




(a) GeoMNIST (I-model).
(b) GeoMNIST (L-model).
(c) CIFAR (I-model).
(d) CIFAR (L-model).

Figure 5: **Relative difference $\mathrm{RD}_k$ between genuine and simulated dynamics.** Note that the L-model performs better than I-model throughout training and better captures the later stages of the training (indicated by decreasing RD).

Penn
UNIVERSITY *of* PENNSYLVANIA

# Contents

# Take-Home Messages

- A phenomenological approach: modeling feature dynamics via SDEs that encodes local elasticity. LE-SDE/ODE can model feature dynamics reasonably well; but to close the gap, we may need to go beyond linearity.
- LE is important for separation of features.
- The LE-SDE can be used to imitate the true dynamics once the LE strengths are estimated.

# Future Works

- **General LE Matrix.** A similar result as in Theorem 1 may be expected for symmetric but no necessarily semi-definite LE matrices $\boldsymbol{E}(t)$.
- **Mini-batch Training, Imbalanced Datasets, and Label Corruptions.** Generalizing the drift matrix to $\boldsymbol{M}_t = (\boldsymbol{E}_t \otimes \boldsymbol{P}) \circ \boldsymbol{H}/K$ for a $K$-by-$K$ doubly stochastic matrix $\boldsymbol{P}$ can be used to model various sampling effects.
- **Beyond L-model for Imitating Genuine Dynamics of DNNs.** Although the L-model is shown to be able to mimic the real dynamics reasonably well, we postulate that a more precise model might have its $(i, j)$-th block encode the other directions other than $\boldsymbol{d}_j$.
- **Finer-Grained Analysis and the Covariance Structure.**
- **Two-Stage Behavior and Exit-Time Analysis.**

Penn
University *of* Pennsylvania

# Acknowledgements

This work was supported in part by NSF through CCF-1934876, an Alfred Sloan Research Fellowship, the Wharton Dean's Research Fund, and ONR Contract N00014-19-1-2620. We would like to thank Dan Roth and the Cognitive Computation Group at the University of Pennsylvania for stimulating discussions and for providing computational resources.

# References


- Shuxiao Chen, Hangfeng He, and Weijie Su, *Label-aware neural tangent kernel: Toward better generalization and local elasticity*, Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 15847–15858.
- Zhun Deng, Hangfeng He, and Weijie Su, *Toward better generalization bounds with locally elastic stability*, Proceedings of the 38th International Conference on Machine Learning, Proceedings of Machine Learning Research, vol. 139, 2021, pp. 2590–2600.
- C. Fang, H. He, Q. Long, and W. J. Su, *Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training*, Proceedings of the National Academy of Sciences (2021).
- Hangfeng He and Weijie Su, *The local elasticity of neural networks*, International Conference on Learning Representations, 2020.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, *Imagenet classification with deep convolutional neural networks*, Advances in Neural Information Processing Systems **25** (2012), 1097–1105.
- Vardan Papyan, X. Y. Han, and David L. Donoho, *Prevalence of neural collapse during the terminal phase of deep learning training*, Proceedings of the National Academy of Sciences **117** (2020), no. 40, 24652–24663.